


Сетевые технологии

Основные принципы



Петухов Андрей
petand@lvk.cs.msu.su
Антоненко Виталий
anvial@lvk.cs.msu.su
комната 247

Стеки протоколов

Необходимость многоуровневости

- Развитие и использование сложных систем без введения уровней абстракции было бы невозможно
- Пример: работа с автомобилем. Уровни абстракции:
 - технологический (завод)
 - инженерный (сервис)
 - пользовательский (водители)
- Пример: работа с РС. Уровни абстракции:
 - аппаратный (электроника)
 - системный (ОС, драйверы)
 - прикладной (программы)
- Появление новых уровней абстракции является следствием (и целью!) развития технологий

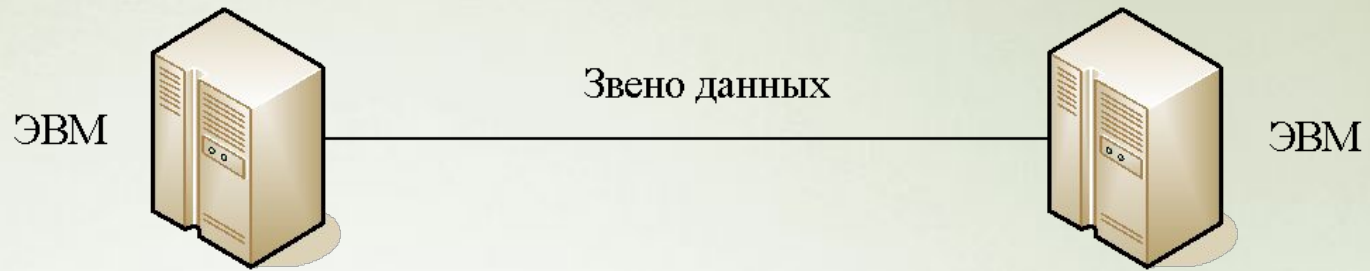
Особенности сетей как предмета деятельности

- Масштабность (большая размерность задач)
- Сложность
- Стохастичность
- Многопараметричность / многокритериальность (Производительность - Качество - Стоимость)
- Многотехнологичность (PSTN - ISDN - Internet - ATM...)
- Инерционность развития

Основные требования к сетям СВЯЗИ

- Эффективность (в смысле бизнес-управления)
- **Расширяемость**
- **Масштабируемость**
- Надежность
- Эксплуатационная пригодность
- Необходимая производительность
- Соответствие стандартам
- Разнообразии обеспечиваемых услуг
(оборудование и сеть = “service enabler”)

Первые системы обмена данными



а) Связь двух ЭВМ



б) Связь терминалов с ЭВМ

Аналог модели OSI



Модель OSI vs модель TCP/IP

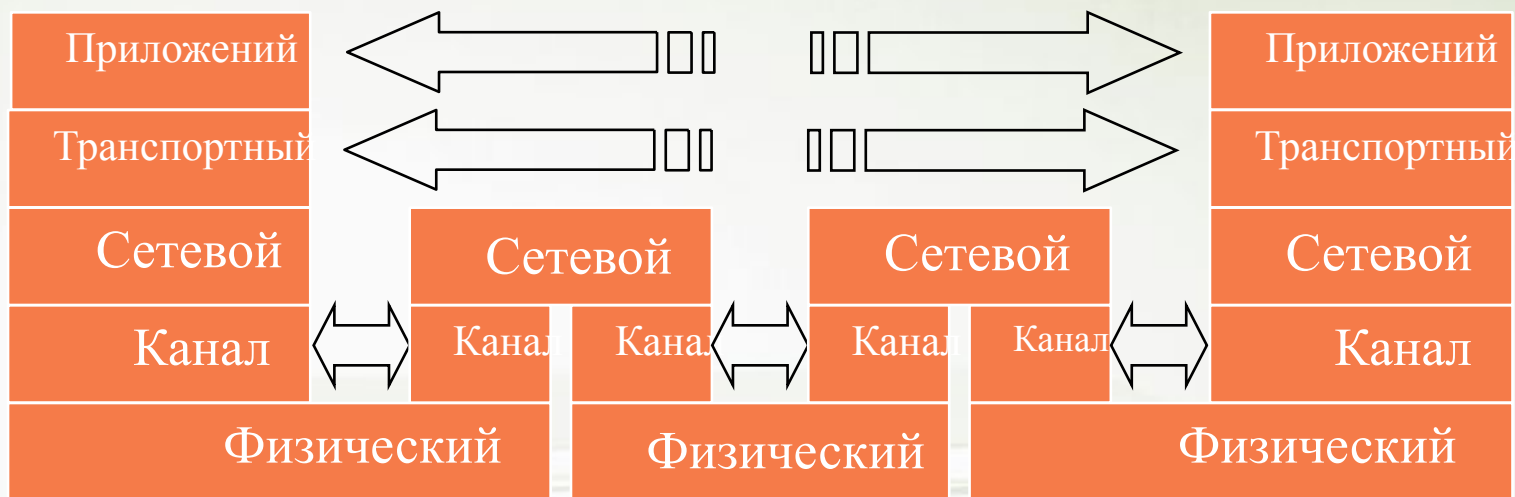


Уровни абстракции в сетевых технологиях

Физический и канальный уровни

- Начальный уровень – умение передавать полезные сигналы в простейших информационных топологиях
 - вещательная топология – радио
 - топология точка-точка – телеграф
- Неизбежность: усложнение топологии
- Необходимость адресации абонентов
- Потребность утвердить формат передачи данных
 - надо отделять пользовательские данные от служебных
 - надо научиться выделять в потоке битов порции данных
- Сервис канального уровня - передать абоненту X из локальной топологии заданную порцию данных
- Уровень создает абстракцию над:
 - деталями форматирования данных
 - алгоритмами доступа к каналу
 - методом кодирования данных в сигналы и т.п.

Модель взаимодействия: ТСР/IP

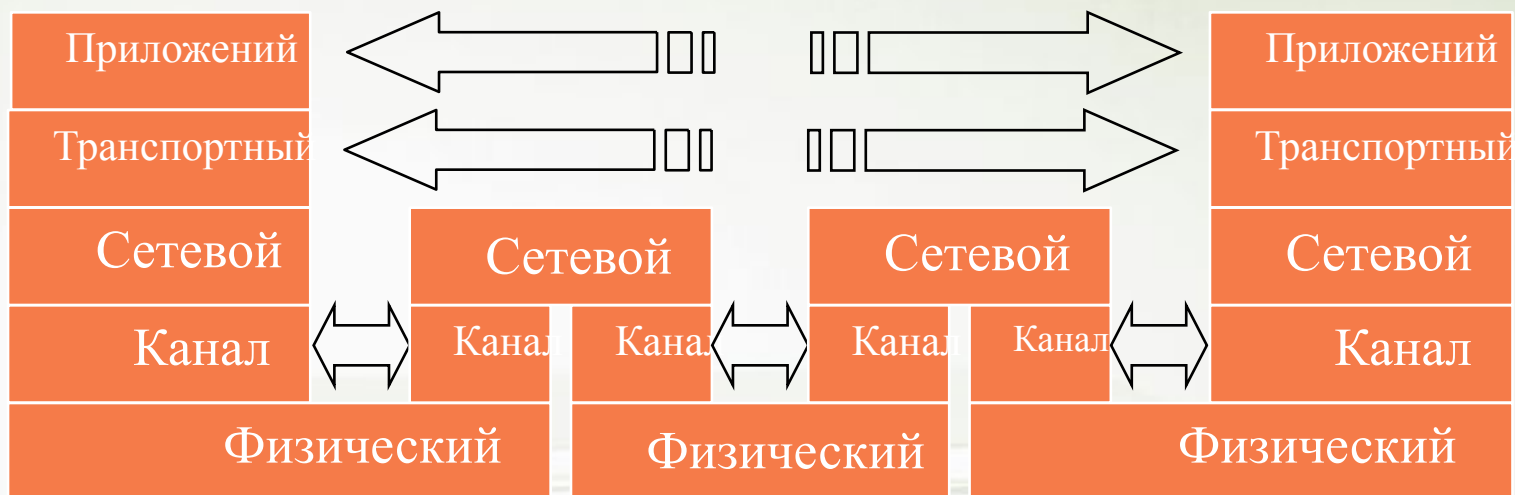


Канальный уровень

Проблема масштабирования

- Устройство канального уровня немного подробнее
 - используемые сетевые устройства – коммутаторы
 - адреса абонентов – адреса, вшитые в сетевые карты
 - уникальность адресов обеспечивается производителями карт
- Таблица коммутации:
 - адрес назначения => номер исходящего порта
- Устремим кол-во абонентов канального уровня к $7 \cdot 10^9$
 - таблица коммутации будет расти линейно
 - скорость поиска исходящего порта в таблице коммутации будет расти линейно
 - следствие - проблема масштабируемости
- Именно эту проблему предлагается решать на следующем уровне

Канальный уровень

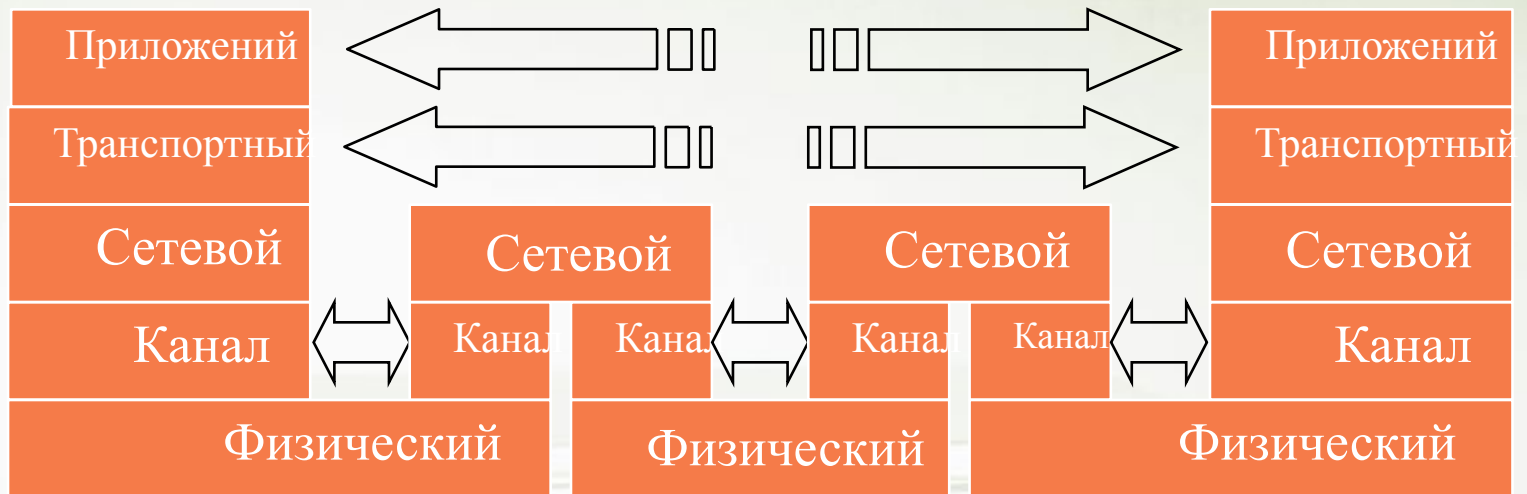


Сетевой уровень

Абстракция над деталями маршрутизации в глобальной сети

- Реализуем иерархичность (как в телефонных сетях)
 - сеть – множество соседних узлов
 - у сети есть глобально уникальный адрес
 - у узла есть адрес, уникальный в пределах сети
 - глобальный адрес узла = (адрес сети, локальный адрес узла)
- Новый уровень назначим ответственным за пересылку данных между сетями (используется адрес сети)
 - за пересылку данных внутри сети назначим ответственным канальный уровень (он же это и умеет!)
- Сервис уровня – пересылка данных между узлами по глобальной сети
- Можно вводить еще уровни иерархии!
 - см. автономные системы

Сетевой уровень

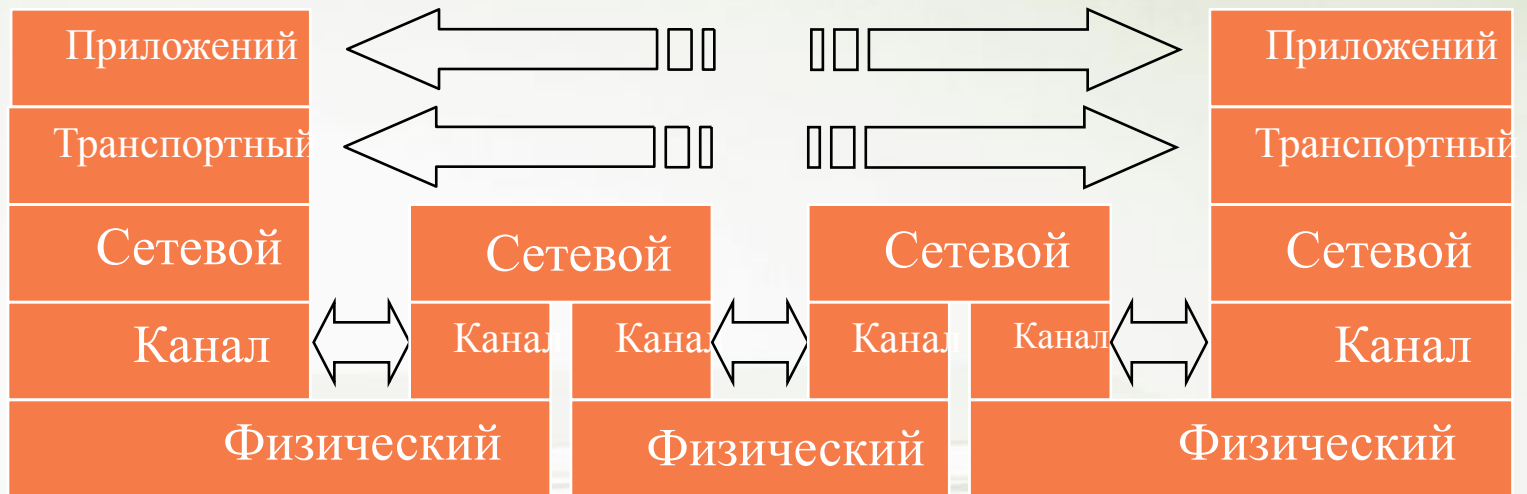


Транспортный уровень

Абстракция над деталями реализации надежного транспорта

- Сервис сетевого уровня адресует узлы целиком
 - на каждом узле может быть много отправителей и получателей
- Отправители хотят получать от сети либо быстрый транспорт, либо надежный
 - странно каждый раз реализовывать свой надежный протокол передачи данных, не так ли?
- Сервис уровня - передача данных между приложениями на разных узлах либо быстро, либо надежно
- Адрес = глобальный адрес узла + адрес приложения
 - адрес сети + локальный адрес узла в сети + адрес приложения на узле

Транспортный уровень

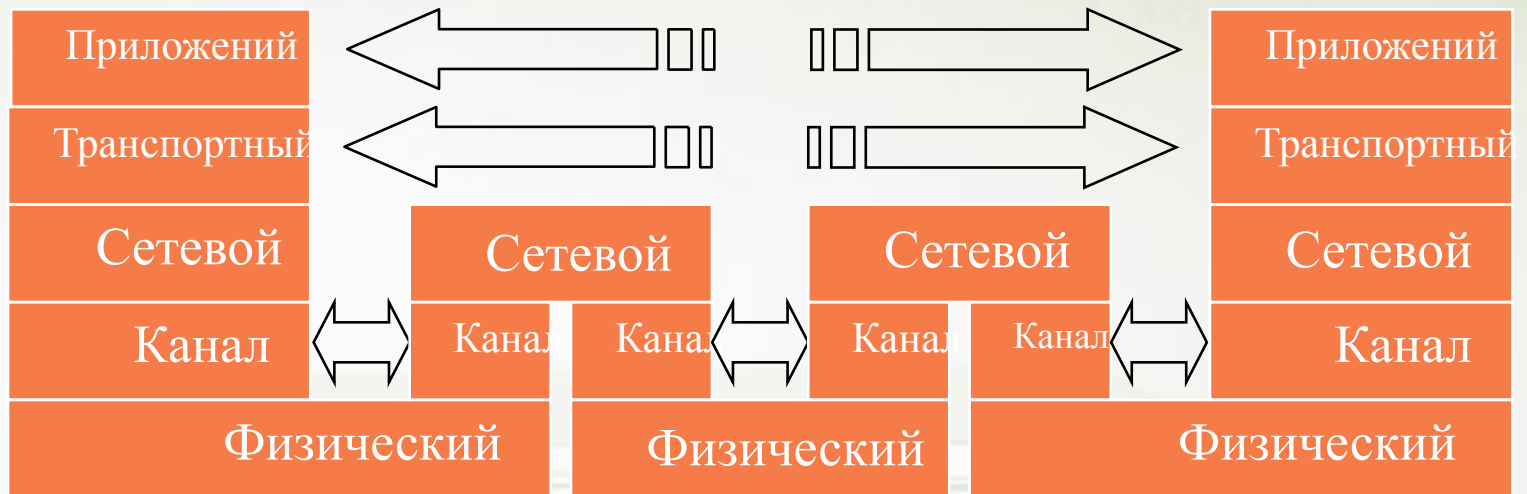


Прикладной уровень

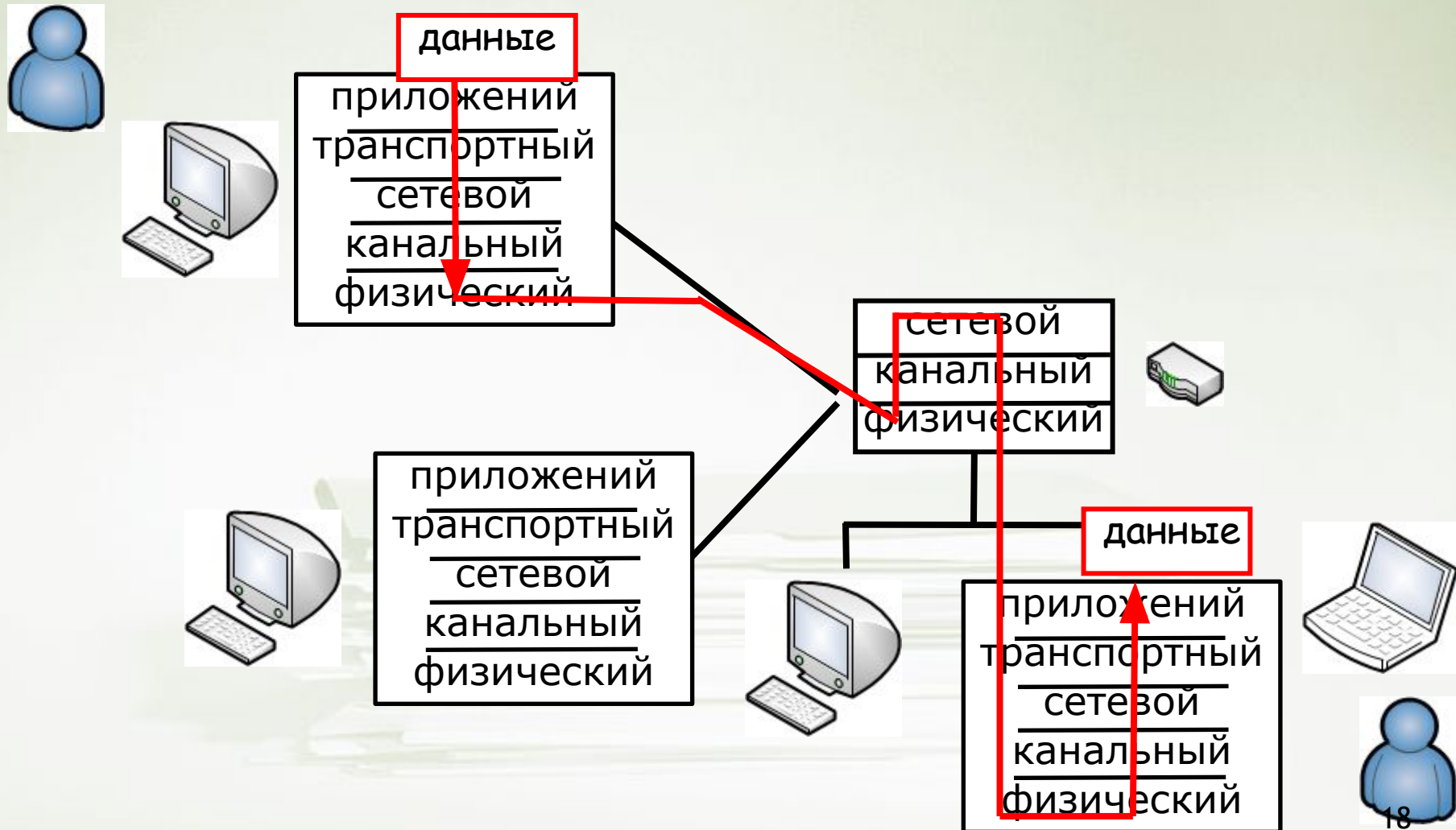
Finally! Решение конкретных прикладных задач

- Доступ к файлам (FTP)
- Передача почты, доступ к почте (SMTP/POP/IMAP)
- Сёрфинг по WWW (HTTP)
- Удаленное администрирование (Telnet/SSH/SNMP)
- Обмен мгновенными сообщениями (IRC)
- Видео- и аудио- трансляции (RTP)
- Адреса – удобные для запоминания человеком
 - DNS

Прикладной уровень



Модель взаимодействия: Пример передачи данных



Кадр, Сегмент, Пакет

- Разное наименование единиц передачи данных на различных уровнях:
 - Канальный уровень - кадр
 - Сетевой уровень - сетевой пакет
 - Транспортный уровень - сегмент
- Данные термины не строго определены
 - В нашем курсе мы «частенько» будем говорить *пакет* независимо от уровня

Типичные задачи

- Задача адресации
- Задача построения таблицы пересылки (коммутации, маршрутизации)
- Задача разрешения адресов верхнего уровня на адреса нижнего уровня
 - DNS => IP
 - IP => MAC



Вопросы?

